易事特集团股份有限公司	文件编号		文件版本	V1.7
	文件密级	非密	生效日期	2020.09
	制定部门	软件部		

EA660 G3 20-250K

Modbus 通信协议

本文件版权属易事特集团股份有限公司所有，违权必究

易事特集团股份有限公司	文件编号		文件版本	V1.7
	文件密级	非密	生效日期	2020.09
	制定部门	软件部		

序号	版本	修改内容	修改人	修改时间	备注
1	Ver 1.0	确定基本的电气量		2012-4-6	
2	Ver 1.1	修改 04 功能代码		2012-09-01	
3	Ver 1.2	修改 02 功能码		2014-04-07	
4	Ver 1.3	修改 02 功能码(适用于监控 V3.6 版本以上)		2015-06-16	
5	Ver 1.4	02 功能码增加		2015-11-05	
6	Ver 1.5	1、04 添加公司信息 2、06/0x10 添加序列号设置 3、删除 ASCII 模式的举例		2019-11-18	
7	Ver 1.6	修改 CRC 校验说明		2020-07-24	
8	Ver 1.7	补充适用范围和接口定义	caolf	2020-09-18	

本文件版权属易事特集团股份有限公司所有，违权必究

目 录

本文件版权属易事特集团股份有限公司所有，违权必究

一、协议相关说明

1、协议适用范围

本协议文档规范了设备 EA660G3 20-250k UPS 通过 RS485 连接提供的 Modbus 接口需求。

2、规范性引用文件

1. RFC791，互联网协议，Sep81 DARPA

2. MODBUS 协议参考指南 Rev J,MODICON，1996 年 6 月，doc#PI_MBUS_300

MODBUS 是一项应用层报文传输协议，用于在通过不同类型的总线或网络连接的设备之间的客户机/服务器通信。目前，使用下列情况实现 MODBUS：

1) 以太网上的 TCP/IP。

2) 各种媒体（有线：EIA/TIA-232-E、EIA/TIA-485-A；光纤、无线等等）上的异步串行传输。

3、协议简介

Modbus 协议是应用于控制器上的一种通用语言。通过该协议使控制器经由网络和其他UPS 设备之间可以进行通信。本通信采用应答方式，由主机发起请求（发送遥测、遥信信息），从机执行请求并且应答。从机需通过地址设置加以区分，从机可设置的地址范围为1~247。

4、接口方式

1、接口说明：

RS485 接口：	异步，半双工
波特率：	默认 9600bps，可设置为1200bps、2400 bps、4800 bps、9600bps
数据长度：	RTU 模式时为8 位
奇偶校验位：	可设置为奇校验、偶校验或者无校验
停止位：	1 位

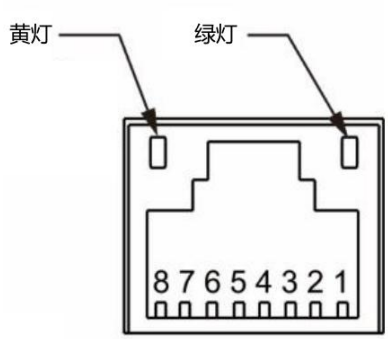
本文件版权属易事特集团股份有限公司所有，违权必究

2、接口示意图

1) 插针端口（2PIN）

针脚（从左往右）	1	2
定义	A 端	B 端

2) RJ45 通讯接口



针脚	1、2	4、5	其他
定义	A 端	B 端	空脚

本文件版权属易事特集团股份有限公司所有，违权必究

5、Modbus RTU 数据帧格式

本协议仅支持Modbus 通信RTU 模式。

控制器以RTU 模式在Modbus 总线上进行通讯时，信息中的每个字节按十六进制。RTU 模式中每个字节的格式为：

编码系统： 8 位二进制；

起始位： 1 位；

数据位： 8 位；

奇/偶校验： 奇校验或者偶校验时为1 位；无奇偶校验时该位为1 位停止位；

停止位： 1 位；

错误校验区：循环冗余校验(CRC)；

RTU 模式的请求帧格式为：

起始	设备地址	功能代码	寄存器起始地址	寄存器个数	CRC 低字节	CRC 高字节	结束
至少3.5 个字符空闲时间	1 byte	1 byte	2 bytes	2 bytes	1 byte	1 byte	至少3.5 个字符空闲时

其中 RTU 模式字符传输格式采用 11 位传输，其中数据位为 8 位，若无奇偶校验位，则 10 位传输

起始位	1	2	3	4	5	6	7	8	停止位（奇/偶校验位）	停止位

RTU 模式的响应帧格式为：

起始	设备地址	功能代码	数据	CRC 低字节	CRC 高字节	结束
至少3.5 个字符空闲时间	1 byte	1 byte	N bytes	1 byte	1 byte	至少3.5 个字符空闲时间

消息发送至少需要 3.5 个字符时间的停顿间隔开始。在最后一个传输字符之后，需要至少 3.5 字符时间的停顿来标定消息的结束。一个新的消息可在此停顿后开始。

整个消息帧必须作为一连续的流转输。如果在帧完成之前两个字符间有超过 1.5 个字符空闲的停顿时间，认为帧错误，停止接收，并重新启动接收。也就是要保证两个帧间的间隔至少大于 3.5 个字符的时间，1.5 个字符时间和 3.5 个字符时间与具体的通信波特率有关，计算方法如下： 如通信波特率为 9600，那么

1.5 个字符间隔时间 =（1/9600）×11×1.5×1000 = 1.72 ms

3.5 个字符间隔时间 =（1/9600）×11×3.5×1000 = 4.01 ms

本文件版权属易事特集团股份有限公司所有，违权必究

【例如】请求1 号机的数据，位置为：寄存器起始地址0002，寄存器个数为1 个，寄存器内容为0x1222

请求帧信息：

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x01	0x03	0x00	0x02	0x00	0x01	0x25	0xCA
字节数	1	1	2		2		2	

响应帧信息：

	地址	功能码	返回数据字节数	数据内容		CRC 校验	
数据	0x01	0x03	0x02	0x12	0x22	0x34	0xFD
字节数	1	1	1	2		2	

6、响应信息分类

主机向从机设备发送查询并希望有一个正常响应，主机查询中有可能产生4 种事件：

（1）从机接收查询，无通讯错误，正常处理信息，则返回一个正常响应事件。

（2）由于通讯出错，从机不能接收查询数据，因而不返回响应。此时，主机依靠处理程序判定为查询超时。

（3）若从机接收查询，发现有 CRC 通讯错误，不返回响应，此时依靠主机处 理程序判定为查询超时。

（4）从机接收查询，无通讯错误，但无法处理(如读不存在的寄存器地址或错误的寄存器个数)时，向主机报告错误的性质。

向主机报告错误的响应信息有2 个与正常响应不相同的区域：

功能代码区：正常响应时，从机的响应功能代码区，带原查询的功能代码。所有功能代码的MSB 为0(其值低于80H)。不正常响应时，从机把功能代码的MSB 置为1，使功能代码值大于 80H，高于正常响应的值。这样，主机应用程序能识别不正常响应事件，能检查不正常代码 的数据区。

数据区：正常响应中，数据区含有(按查询要求给出的)数据或统计值，在不正常响应中，数据区为一个不正常代码，它说明从机产生不正常响应的条件和原因。

本文件版权属易事特集团股份有限公司所有，违权必究

不正常代码及含义如下表所示：

代码	名称	含义
0x01	不合法功能代码	从机接收的是一种不能执行功能代码。发出查询命令后，该代码指示无程序功能
0x02	不合法数据地址	接收的数据地址，是从机不允许的地址。
0x03	不合法数据	查询数据区的值是从机不允许的值。
0x04	从机设备故障	从机执行主机请求的动作时出现不可恢复的错误。
0x08	内存奇偶校验错误	从机读扩展内存中的数据时，发现有奇偶校验错误，主机按从机的要求重新发送数据请求。

【例如】***

RTU 模式：（ASCII 模式类似）

命令信息：请求1 号机的数据，位置为：寄存器起始地址0066，寄存器个数为2 个

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x01	0x03	0x00	0x66	0x00	0x02	0x24	0x14

响应信息：1 号机的响应帧，因为寄存器起始地址错误，因此返回信息为不合法的数据地址

	地址	功能码	数据内容	CRC 校验	
数据	0x01	0x83	0x02	0xC0	0xF1

7、功能代码

功能码	名称	作用
0x02	读离散量输入	读从机离散量输入寄存器中的二进制数据
0x04	读取输入寄存器	在一个或多个输入寄存器取得当前的二进制值

本文件版权属易事特集团股份有限公司所有，违权必究

二、寄存器列表

1、读输入寄存器（功能码 0x04）

地址		数据内容	数据长度/格式	类型	单位	系数	备注
HEX	DEC						
0x0000	0	版本号	2bytes	USHORT	1	0.1	程序版本号
0x0001	1	R 相输入电压	2bytes	USHORT	V	1	
0x0002	2	S 相输入电压	2bytes	USHORT	V	1	
0x0003	3	T 相输入电压	2bytes	USHORT	V	1	
0x0004	4	R 相输入频率	2bytes	USHORT	Hz	0.1	
0x0005	5	S 相输入频率	2bytes	USHORT	Hz	0.1	
0x0006	6	T 相输入频率	2bytes	USHORT	Hz	0.1	
0x0007	7	R 相旁路电压	2bytes	USHORT	V	1	
0x0008	8	S 相旁路电压	2bytes	USHORT	V	1	
0x0009	9	T 相旁路电压	2bytes	USHORT	V	1	
0x000A	10	R 相旁路频率	2bytes	USHORT	Hz	0.1	
0x000B	11	S 相旁路频率	2bytes	USHORT	Hz	0.1	
0x000C	12	T 相旁路频率	2bytes	USHORT	Hz	0.1	
0x000D	13	R 相输出电压	2bytes	USHORT	V	0.1	
0x000E	14	S 相输出电压	2bytes	USHORT	V	0.1	
0x000F	15	T 相输出电压	2bytes	USHORT	V	0.1	
0x0010	16	R 相输出电流	2bytes	USHORT	A	0.1	
0x0011	17	S 相输出电流	2bytes	USHORT	A	0.1	
0x0012	18	T 相输出电流	2bytes	USHORT	A	0.1	
0x0013	19	R 相输出频率	2bytes	USHORT	Hz	0.1	
0x0014	20	S 相输出频率	2bytes	USHORT	Hz	0.1	
0x0015	21	T 相输出频率	2bytes	USHORT	Hz	0.1	
0x0016	22	R 相有功功率	2bytes	USHORT	KW	1	
0x0017	23	S 相有功功率	2bytes	USHORT	KW	1	
0x0018	24	T 相有功功率	2bytes	USHORT	KW	1	
0x0019	25	R 相视在功率	2bytes	USHORT	KVA	1	
0x001A	26	S 相视在功率	2bytes	USHORT	KVA	1	
0x001B	27	T 相视在功率	2bytes	USHORT	KVA	1	
0x001C	28	R 相负载率	2bytes	USHORT	%	1	
0x001D	29	S 相负载率	2bytes	USHORT	%	1	
0x001E	30	T 相负载率	2bytes	USHORT	%	1	
0x001F	31	正充电电压	2bytes	USHORT	V	1	
0x0020	32	负充电电压	2bytes	USHORT	V	1	
0x0021	33	正充电电流	2bytes	USHORT	A	0.1	
0x0022	34	负充电电流	2bytes	USHORT	A	0.1	
0x0023	35	充电器温度	2bytes	SHORT	℃	0.1	
0x0024	36	正电池电压	2bytes	USHORT	V	1	

0x0025	37	负电池电压	2bytes	USHORT	V	1	
0x0026	38	电池容量	2bytes	USHORT	%	1	
0x0027	39	剩余放电时间	2bytes	USHORT	min	1	
0x0028	40	电池温度 1	2bytes	SHORT	℃	1	
0x0029	41	电池温度 2	2bytes	SHORT	℃	1	
0x002A	42	电池温度 3	2bytes	SHORT	℃	1	
0x002B	43	电池温度 4	2bytes	SHORT	℃	1	
0x002C	44	模块温度	2bytes	SHORT	℃	0.1	
0x002D	45	工作模式	2bytes	USHORT			1：待机模式 2：旁路模式 3：市电模式 4：电池模式 5：电池自检 6：故障模式 7：变频模式 8：紧急关机模式 9：关机模式
0x002E	46	模块 1~16 存在标志	2bytes	USHORT			第 N 个模块 存在 1<<N-1
0x002F	47	模块 17~32 存在标志	2bytes	USHORT			第 N 个模块 存在 1<<N-1
0x0030	48	模块 1~16 故障标志	2bytes	USHORT			第 N 个模块 故障 1<<N-1
0x0031	49	模块 17~32 故障标志	2bytes	USHORT			第 N 个模块 故障 1<<N-1
0x0032	50	模块 1~16 告警标志	2bytes	USHORT			第 N 个模块 告警 1<<N-1
0x0033	51	模块 17~32 告警标志	2bytes	USHORT			第 N 个模块 告警 1<<N-1
0x0034	52	预留	2bytes	/			
0x0035	53	预留	2bytes	/			
0x0036	54	预留	2bytes	/			

2、读离散量（功能码0x02）

地址		告警/故障	数据长度/格式	说明
HEX	DEC			
0x0000	0	模块 BUS 高压	1 bit	1 表示模块故障发生，0 未发生
0x0001	1	模块 BUS 低压	1 bit	
0x0002	2	模块 BUS 不平衡	1 bit	
0x0003	3	模块 BUS 短路	1 bit	
0x0004	4	（预留）	1 bit	
0x0005	5	模块 BUS 软启超时	1 bit	
0x0006	6	模块逆变软启超时	1 bit	
0x0007	7	模块逆变高压	1 bit	
0x0008	8	模块逆变低压	1 bit	
0x0009	9	模块 R 相输出短路	1 bit	
0x000A	10	模块 S 相输出短路	1 bit	
0x000B	11	模块 T 相输出短路	1 bit	
0x000C	12	模块 RS 相输出短路	1 bit	
0x000D	13	模块 ST 相输出短路	1 bit	
0x000E	14	模块 TR 相输出短路	1 bit	
0x000F	15	模块 R 相输出负功异常	1 bit	
0x0010	16	模块 S 相输出负功异常	1 bit	
0x0011	17	模块 T 相输出负功异常	1 bit	
0x0012	18	模块过载故障	1 bit	
0x0013	19	模块电池异常	1 bit	
0x0014	20	预留	1 bit	
0x0015	21	模块过温故障	1 bit	
0x0016	22	模块同步信号故障	1 bit	
0x0017	23	模块同步脉冲故障	1 bit	
0x0018	24	模块继电器粘死	1 bit	
0x0019	25	市电 SCR 故障	1 bit	
0x001A	26	模块 CAN 总线故障	1 bit	
0x001B	27	模块总负功故障	1 bit	
0x001C	28	模块物理地址冲突	1 bit	
0x001D	29	模块 IGBT 短路	1 bit	
0x001E	30	逆变器异常	1 bit	
0x001F	31	旁路接线错误	1 bit	
0x0020	32	充电器 BUS 高压	1 bit	1 表示充电器故障发生，0 未发生

本文件版权属易事特集团股份有限公司所有，违权必究

0x0021	33	充电器 BUS 低压	1 bit	
0x0022	34	充电器 BUS 不平衡	1 bit	
0x0023	35	充电器 BUS 短路	1 bit	
0x0024	36	充电器 BUS 软启超时	1 bit	
0x0025	37	充电器 Buck 软启超时	1 bit	
0x0026	38	充电器过温	1 bit	
0x0027	39	充电器 Relay 异常	1 bit	
0x0028	40	充电器市电 SCR 异常	1 bit	
0x0029	41	充电器软启继电器异常	1 bit	
0x002A	42	过充自杀	1 bit	
0x002B	43	电池反接	1 bit	
0x002C	44	充电器 ID 错误	1 bit	
0x002D	45	（预留）	1 bit	
0x002E	46	（预留）	1 bit	
0x002F	47	（预留）	1 bit	
0x0030	48	模块未锁（模块）	1 bit	1 表示模块告警发生， 0 未发生
0x0031	49	过载异常（模块）	1 bit	
0x0032	50	通信故障（模块）	1 bit	
0x0033	51	UPS 过载（模块）	1 bit	
0x0034	52	电池未接（模块）	1 bit	
0x0035	53	保留	1 bit	
0x0036	54	UPS 过流（模块）	1 bit	
0x0037	55	电池电压异常（模块）	1 bit	
0x0038	56	冗余过载（模块）	1 bit	
0x0039	57	读写 EEROM 错误（模块）	1 bit	
0x003A	58	模块风扇故障（模块）	1 bit	
0x003B	59	市电相序错误（模块）	1 bit	
0x003C	60	旁路相序错误（模块）	1 bit	
0x003D	61	N 线未接（模块）	1 bit	
0x003E	62	同步信号故障（模块）	1 bit	
0x003F	63	同步脉冲故障（模块）	1 bit	
0x0040	64	市电异常（模块）	1 bit	
0x0041	65	旁路异常（模块）	1 bit	
0x0042	66	电池低压（模块）	1 bit	
0x0043	67	保留	1 bit	
0x0044	68	模块未检测到充电器（模块）	1 bit	
0x0045	69	保留 5（模块）	1 bit	
0x0046	70	旁路频率异常（模块）	1 bit	
0x0047	71	输出过压（模块）	1 bit	

本文件版权属易事特集团股份有限公司所有，违权必究

0x0048	72	物理地址冲突（模块）	1 bit	
0x0049	73	R 相 PFC 异常（模块）	1 bit	
0x004A	74	S 相 PFC 异常（模块）	1 bit	
0x004B	75	T 相 PFC 异常（模块）	1 bit	
0x004C	76	输出异常（模块）	1 bit	
0x004D	77	旁路 STS 异常（模块）	1 bit	
0x004E	78	旁路 STS 短路（模块）	1 bit	
0x004F	79	预留	1 bit	
0x0050	80	电池过充(充电器)	1 bit	1 表示充电器告警发生， 0 未发生
0x0051	81	充电器输出保险断开	1 bit	
0x0052	82	电池未接(充电器)	1 bit	
0x0053	83	充电器异常	1 bit	
0x0054	84	电池高温(充电器)	1 bit	
0x0055	85	充电器未开	1 bit	
0x0056	86	充电器输出 SCR 异常	1 bit	
0x0057	87	电池温度异常(充电器)	1 bit	
0x0058	88	传感器异常(充电器)	1 bit	
0x0059	89	LCD 掉线(充电器)	1 bit	
0x005A	90	充电电压设置错误(充电器)	1 bit	
0x005B	91	充电电流设置错误(充电器)	1 bit	
0x005C	92	充电模块未锁	1 bit	
0x005D	93	充电器风扇故障	1 bit	
0x005E	94	市电相序错误(充电器)	1 bit	
0x005F	95	N 线未接(充电器)	1 bit	
0x0060	96	模块不兼容	1 bit	1 标示模块新增故障发生 0 未发生
0x0061	97	AD 采样异常	1 bit	
0x0062	98	直流分量异常	1 bit	
0x0063	99	预留	1 bit	
0x0064	100	预留	1 bit	
0x0065	101	预留	1 bit	
0x0066	102	预留	1 bit	
0x0067	103	预留	1 bit	
0x0068	104	预留	1 bit	
0x0069	105	预留	1 bit	
0x006A	106	预留	1 bit	
0x006B	107	预留	1 bit	
0x006C	108	预留	1 bit	
0x006D	109	预留	1 bit	

本文件版权属易事特集团股份有限公司所有，违权必究

0x006E	110	预留	1 bit	
0x006F	111	预留	1 bit	

本文件版权属易事特集团股份有限公司所有，违权必究

三、通信内容

1、读输入寄存器（功能码0x04）

【举例】

假设UPS 设备地址设置为0x18，查询寄存器起始地址值为0x0010，寄存器个数为2 个，即查询“R 相输出电流”和“S 相输出电流”的值；假设此时“R 相输出电流”的值为89.2A，“S 相输出电流”的值为88.9A，根据该值的系数为0.1，那么：

寄存器0x0010 的值为：$(892)_D = (037C)_H$

寄存器0x0011 的值为：$(889)_D = (0379)_H$

则返回数据的字节数为4 个，RTU 模式时，对数据查询的请求帧信息和响应帧信息为：

请求帧信息为：

	地址	功能码	寄存器起始地址		寄存器个数		CRC 校验	
数据	0x18	0x04	0x00	0x10	0x00	0x02	0x72	0x07
字节数	1	1	2		2		2	

响应帧信息为：

	地址	功能码	返回数据字节数	数据内容		CRC 校验	
数据	0x18	0x04	0x04	0x037C	0x0379	0x73	0xCB
字节数	1	1	1	4		2	

本文件版权属易事特集团股份有限公司所有，违权必究

2、读离散量（功能码0x02）

【举例】

假设UPS 设备地址设置为0x18，查询寄存器起始地址值为51，即0x0033，寄存器个数为1个，即查询“UPS 过载状态”；假设此时“UPS 已过载”；即该值为1。 返回数据时，在该字节中由低位向高位排列，直至8 个位为止。下一个字节中的8 个输入位也是从低位到高位排列。若返回的输入位数不是8 的倍数，则在最后的数据字节中的剩余位直至 字节的最高位全部填零。RTU 模式时，对状态查询的请求帧信息和响应帧信息为：

请求帧信息为：

	地址	功能码	起始地址		离散量个数		CRC 校验	
数据	0x18	0x02	0x00	0x33	0x00	0x01	0x4B	0xCC
字节数	1	1	2		2		2	

响应帧信息为：

	地址	功能码	返回数据字节数	数据内容	CRC 校验	
数据	0x18	0x02	0x01	0x01	0x67	0x14
字节数	1	1	1	1	2	

本文件版权属易事特集团股份有限公司所有，违权必究

附录A　　CRC 校验

CRC 循环冗余校验

循环冗余校验 (CRC) 域为两个字节，包含一个二进制 16 位值。附加在报文后面的 CRC 的值由发送设备计算。接收设备在接收报文时重新计算 CRC 的值，并将计算结果于实际接收到的 CRC 值相比较。如果两个值不相等，则为错误。

CRC 的计算, 开始对一个 16 位寄存器预装全 1. 然后将报文中的连续的 8 位子节对其进行后续的计算。只有字符中的 8 个数据位参与生成 CRC 的运算，起始位，停止位和校验位不参与 CRC 计算。CRC 的生成过程中， 每个 8–位字符与寄存器中的值异或。然后结果向最低有效位 (LSB) 方向移动(Shift) 1 位，而最高有效位 (MSB) 位置充零。 然后提取并检查 LSB：如果 LSB 为 1，则寄存器中的值与一个固定的预置值异或；如果 LSB 为 0， 则不进行异或操作。这个过程将重复直到执行完 8 次移位。完成最后一次（第 8 次）移位及相关操作后，下一个 8 位字节与寄存器的当前值异或，然后又同上面描述过的一样重复 8 次。当所有报文中子节都运算之后得到的寄存器中的最终值，就是 CRC。

生成 CRC 的过程为:

1. 将一个 16 位寄存器装入十六进制 FFFF (全 1). 将之称作 CRC 寄存器.

2. 将报文的第一个 8 位字节与 16 位 CRC 寄存器的低字节异或，结果置于 CRC 寄存器.

3. 将 CRC 寄存器右移 1 位 (向 LSB 方向)， MSB 充零. 提取并检测 LSB.

4. (如果 LSB 为 0): 重复步骤 3 (另一次移位).

(如果 LSB 为 1): 对 CRC 寄存器异或多项式值 0xA001 (1010 0000 0000 0001).

5. 重复步骤 3 和 4，直到完成 8 次移位。当做完此操作后，将完成对 8 位字节的完整操作。

6. 对报文中的下一个字节重复步骤 2 到 5，继续此操作直至所有报文被处理完毕。

7. CRC 寄存器中的最终内容为 CRC 值.

8. 当放置 CRC 值于报文时，如下面描述的那样，高低字节必须交换。

将 CRC 放置于报文当 16 位 CRC (2 个 8 位字节) 在报文中传送时，低位字节首先发送，然后是高位字节。

例:执行 CRC 生成的 C 语言的函数在下面示出。所有的可能的 CRC 值都被预装在两个数组中，当计算报文内容时可以简单的索引即可。一个数组含有 16 位 CRC 域的所有 256 个可能的高位字节，另一个数组含有低位字节的值。这种索引访问 CRC 的方式提供了比对报文缓冲区的每个新字符都计算新的 CRC 更快的方法。

注意: 此函数内部执行高/低 CRC 字节的交换。此函数返回的是已经经过交换的 CRC 值。

本文件版权属易事特集团股份有限公司所有，违权必究

也就是说，从该函数返回的 CRC 值可以直接放置于报文用于发送。

函数使用两个参数:

unsigned char *puchMsg;指向含有用于生成 CRC 的二进制数据报文缓冲区的指针

unsigned short usDataLen; 报文缓冲区的字节数.

CRC 生成函数

```
unsigned short CRC16 ( puchMsg， usDataLen ) /* 函数以 unsigned short 类型返回 CRC */
unsigned char *puchMsg ; /* 用于计算 CRC 的报文 */
unsigned short usDataLen ; /* 报文中的字节数 */
{
    unsigned char uchCRCHi = 0xFF ; /* CRC 的高字节初始化 */
    unsigned char uchCRCLo = 0xFF ; /* CRC 的低字节初始化 */
    unsigned uIndex ;              /* CRC 查询表索引 */
    while (usDataLen--)            /* 完成整个报文缓冲区 */
    {
        uIndex = uchCRCLo ^ *puchMsgg++ ; /* 计算 CRC */
        uchCRCLo = uchCRCHi ^ auchCRCHi[uIndex] ;
        uchCRCHi = auchCRCLo[uIndex] ;
    }
     return (uchCRCHi << 8 | uchCRCLo) ;
}
```


本文件版权属易事特集团股份有限公司所有，违权必究

附录B　　高低位字节表

高字节表

/* 高位字节的 CRC 值 */

static unsigned char auchCRCHi[] = {

0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0,

0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1,

0x81, 0x40, 0x01,

0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01,

0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,

0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80,

0x41, 0x01, 0xC0,

0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,

0x80, 0x41, 0x01,

0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00,

0xC1, 0x81, 0x40,

0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81,

0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0,

0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1,

0x81, 0x40, 0x01,

0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01,

0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,

0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,

0x40, 0x01, 0xC0,

0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,

本文件版权属易事特集团股份有限公司所有，违权必究

```
0x80, 0x41, 0x01,
0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01,
0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
0x00, 0xC1, 0x81,
0x40
} ;
```

低字节表

```
/* 低位字节的 CRC 值 */
static char auchCRCLo[] = {
0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7,
0x05, 0xC5, 0xC4,
0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB,
0x0B, 0xC9, 0x09,
0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE,
0xDF, 0x1F, 0xDD,
0x1D, 0x1C, 0xDC, 0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2,
0x12, 0x13, 0xD3,
0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32,
0x36, 0xF6, 0xF7,
0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E,
0xFE, 0xFA, 0x3A,
0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38, 0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B,
0x2A, 0xEA, 0xEE,
0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27,
0xE7, 0xE6, 0x26,
0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1,
0x63, 0xA3, 0xA2,
0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4, 0x6C, 0xAC, 0xAD,
0x6D, 0xAF, 0x6F,
0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8,
0xB9, 0x79, 0xBB,
0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4,
```


本文件版权属易事特集团股份有限公司所有，违权必究

```
0x74, 0x75, 0xB5,
0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,
0x50, 0x90, 0x91,
0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94,
0x54, 0x9C, 0x5C,
0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59,
0x58, 0x98, 0x88,
0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D,
0x4D, 0x4C, 0x8C,
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83,
0x41, 0x81, 0x80,
0x40
};
```


本文件版权属易事特集团股份有限公司所有，违权必究